1. Tách từ
   Do tiếng việt không thể dùng khoảng trắng để phân tách từ, hơn nữa trong một số trường hợp tách từ phức tạp như:
   (Học sinh học sinh học.) => (, Học sinh, học, sinh học, .,)
   => Cần phải có mô hình tách từ hiệu quả
   Sử dụng thư viện ViTokenizer
   Link: https://pypi.org/project/pyvi/
   Với độ chính xác 0.985
2. Tokenize
   Sử dụng thư viện Tokenizer()
   Để encode đoạn văn cần phân loại sang vector
   Thư viện này sẽ tạo từ điển bằng phương thức fit.
   Sau khi có từ điển thì sẽ chuyển từ sang số tương ứng trong từ điển đó.
   Link:
   https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/text/Tokenizer
3. Padding vector
   Vector thu được ở trên sẽ có chiều dài khác nhau, ta cần padding để các vector này có cùng độ dài.
   Phương pháp padding:
   - Nếu chưa đủ max len thì ta thêm 0 ở trước.
   - Nếu quá max len thì ta cắt phần sau.

   Thư viện hỗ trợ:
   Link: https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/sequence/pad_sequences
4. Word Embedding
   Để mô hình hiệu quả thì mỗi vector đại diện cho từ sẽ có mối quan hệ gần nhau ví dụ học sinh, sinh viên thì sẽ gần nhau hơn so với học sinh, cầu thủ nhằm tăng độ chính xác của mô hình.
   Để làm điều đó ta sử dụng pretrained model Word Embedding
   Link:
   https://thiaisotajppub.s3-ap-northeast-1.amazonaws.com/publicfiles/baomoi.window2.vn.model.bin.gz
5. Model sử dụng:
   Mô hình sử dụng gồm Embedding layer, Bidirectional LSTM, Dense (activation='softmax' => Đầu ra là xác suất cho từng class)

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_3 (Embedding)      (None, 100, 300)          9052500
_________________________________________________________________
bidirectional_3 (Bidirection (None, 128)               186880
_________________________________________________________________
dense_3 (Dense)              (None, 4)                 516
=================================================================
Total params: 9,239,896
Trainable params: 9,239,896
Non-trainable params: 0
```